

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

8



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
 United States Patent and Trademark Office
 Address: COMMISSIONER FOR PATENTS
 P.O. Box 1450
 Alexandria, Virginia 22313-1450
 www.uspto.gov

2

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/025,774	12/26/2001	Mauricio Lopez	BS01-321	5878

28970 7590 10/05/2004

SHAW PITTMAN
 IP GROUP
 1650 TYSONS BOULEVARD
 SUITE 1300
 MCLEAN, VA 22102

EXAMINER

CHOW, CHIH CHING

ART UNIT PAPER NUMBER

2122

DATE MAILED: 10/05/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

6

Office Action Summary

Application No.

10/025,774

Applicant(s)

LOPEZ ET AL.

Examiner

Chih-Ching Chow

Art Unit

2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 26 December 2001.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-21 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☐ Claim(s) _____ is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 26 December 2001 is/are: a) ☐ accepted or b) ☒ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is responsive to the application filed on December 26, 2001.
2. The priority date considered for this application is December 26, 2001.
3. Claims 1-21 have been examined.

Drawings

4. The drawings are objected to because in FIG 6, 601 should be **604** (see description in paragraph 44). The objection to the drawings will not be held in abeyance.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 5, 13, and 19 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

7. The use of the trademark has been noted in the specification and in claims 5, 13, and 19, **Java RMI**. It should be capitalized wherever it appears and be accompanied by the generic terminology.

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner which might adversely affect their validity as trademarks.

To expedite correction on this matter, the examiner suggests the following guidelines for Applicant to follow in amending the specification:

- a. Capitalize each letter of a trademark or accompany the trademark with an appropriate designation symbol, e.g., TM or ®, as appropriate.
- b. Use each trademark as an adjective modifying a descriptive noun. For example, it would be appropriate to recite "the JAVA platform" or "the JAVA programming language." Note that in these examples, "platform" and "programming language" provide accompanying generic terminology, describing the context in which the trademark is used. By itself, the trademark JAVA specifies only the source of the so-labeled products, namely SUN Microsystems, Inc.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

10. Claims 1-5, 9-13, 17, and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,662,363 by Hiromu Miyamoto (hereinafter "Miyamoto"), in view of 'Introduction to java Remote Method Invocation (RMI)', by Chris Matthews (hereinafter, "Matthews").

CLAIM

1. A method for modifying a computer application in substantially real-time without suspending or terminating the application, said method comprising the steps of:

(a) connecting to an application executing on an application server, the application server having a computer memory;

(b) acquiring program data related to the program structure of the application;

(c) displaying the program data to a maintenance person;

(d) accepting a command from the maintenance person; and

Miyamoto / Matthews

Miyamoto shows the features of claim 1 items a, b, c, d, and e in his art. In Miyamoto, column 4, lines 23-35, "Reference numeral 300 denotes a **server**, which includes a **data transmission/reception section 301** for **transmitting and receiving various data (acquiring program data)** to and from the communication network 200, and a **data storage device (computer memory) 302** for storing software to be supplied to users." See Miyamoto's Fig. 1, a **server** is connecting to a network, and with a **storage device** (items a and b).

For item c and d, in Miyamoto's abstract, "The **display** of the contents of the agreement and determination of the **user's selection (accepting a command from a user)** to agree or to not agree to the contents of the agreement may be performed" and in column 1, lines 26-28, "a predetermined screen for a licensing agreement concerning the software to be installed (hereinafter also called "new software") is caused to show up on a computer

(e) executing the command to cause the executing application to be modified without suspending or terminating the executing application.

display". For item e, Miyamoto's column 1, lines 32-34, "the setup routine waits until one of the "Agree" and "Not Agree" buttons is depressed or activated by the user. If the user has activated the "Agree" button, the setup routine **proceeds to step SP104 to execute the installation of the software".** Miyamoto teaches all aspects of claim 1 but does not mention the 'real-time without terminating while modifying a computer application' specifically. However, Matthews teaches that feature in an analogous art. In Matthews, last paragraph on page 8, "If you are looking at this server application and wondering how **it continues to run** after it has seemingly completed its mission, the answer is that the main thread goes away at this point. However, when the **server** calls the registry to bind the object, it creates another thread under the covers that blocks waiting in a loop for a registry derigstration event. This **keeps the server from terminating.**"

It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Miyamoto's teaching with that of real-time updating computer application taught by Matthews, for the purpose of facilitating the development of cross-platform distributed computer applications. (see Matthews, page 1, 3rd paragraph).

2. The method of claim 1, further comprising the step of modifying data that has been cached in the computer memory.

For the feature of claim 1 see claim 1 rejection. In Miyamoto, column 6, lines 49-53, "the partly-changed software currently **stored in the memory** is automatically executed just as it is." Storing the modifying data into cache memory for faster accessing is a design choice; it's well-known to those skilled in the art.

3. The method recited in claim 1, further comprising the step of modifying an order of the execution of a plurality of methods within the application.

For the feature of claim 1 see claim 1 rejection. In Miyamoto, claim 2 and claim 3, "

2. A method as recited in claim 1 wherein the steps of said method are executed **sequentially by said computer in accordance with a predetermined setup program.**

3. A method as recited in claim 1 wherein said software installation is executed by said computer in accordance with a **predetermined setup program**, and said step of displaying the contents of the agreement, said step of **receiving** and said step of **deleting** are executed by said computer running the installed software." The **predetermined setup program** can contain the 'order' of the execution; the 'receiving' and 'deleting' software is **modifying the order** of the execution.

4. The method recited in claim 1, further comprising the step of providing more detailed diagnostic messages in response to the command.

For the feature of claim 1 see claim 1 rejection. In Miyamoto's claim 1, "determining whether the computer satisfies a predetermined installation

condition; if the predetermined installation condition is not satisfied, **displaying the reasons for failing to** satisfy the predetermined installation condition; if the predetermined installation condition is satisfied, then executing the software installation and determining whether the software installation is successful; if the software installation is determined to be unsuccessful, **displaying** an error indication indicating that the software installation is unsuccessful; if the software installation is determined to be successful, then thereafter **displaying** contents of an agreement concerning the installed software; receiving a user's selection as to whether or not to agree to the contents of the agreement; and deleting the installed software from the computer, when the user's selection has been made to not agree to the contents of the agreement." - all the **displaying** cited above are **diagnostic messages and responses to the commands**.

5. The method recited in claim 1, further comprising the step of connection to the application using Java RMI.

For the feature of claim 1 see claim 1 rejection. In Matthews, the feature of connecting to the application using Java RMI is taught, see Matthews page 1, 4th paragraph, "Java RMI is shipped with the Java JDK 1.1 and higher. It is a true **distributed computing application** interface for Java".

9. A system for modifying an application in substantially real-time during execution without suspending or terminating the application, comprising:

Same as claim 1 rejection.

(a) an application server on which the application executes;

Same as claim 1 (a) rejection.

an object shell console that attaches to the application while it is running

(b) a graphical user interface in the object shell console that is used to assist a maintenance person in modifying the application; and

Same as claim 1 (c) rejection.

(c) a command line for accepting a command to be executed, said command when executed will cause the execution of the application to be modified without suspending or terminating the application.

Same as claim 1 (e) rejection.

10. The system recited in claim 9, further comprising:

For the feature of claim 9 see claim 9 rejection. For the rest of the feature see claim 1 (d), and claim 3 rejection.

a vector for establishing an order of method execution in the application;

a command line for entering a new vector comprising a different order for executing the methods in the application; and

wherein entering the new vector in the command line establishes the different order of method execution.

11. The system recited in claim 9, further comprising a data cache in a memory of the application server, said data cache being modified by the command.

For the feature of claim 9 see claim 9 rejection. For the rest of the feature in claim 11, see claim 2 rejection.

12. The system recited in claim 9, further comprising:
 a terse logging operation that provides terse diagnostic messages;
 a detailed logging operation that provides detailed diagnostic messages;
 and
 means for allowing a maintenance person to select detailed diagnostic messages if an error occurs without suspending or terminating the application.

For the feature of claim 9 see claim 9 rejection. On Matthews, page 3, last paragraph, "The remote interface needs to import the RMI package, and every exported method must throw an RMI remote exception to **manage errors** during invocation", Mathews also teaches a terse error reporting in his sample programs on page 7 and page 8. In Miyamoto's claim 1, "if the software installation is determined to be unsuccessful, **displaying an error indication** indicating that the software installation is unsuccessful;" Miyamoto teaches to log detailed diagnostic messages (also see claim 4 rejection). Since Matthews uses the RMI package, and RMI allows modifying application without suspending or terminating the application, therefore no suspension or termination will be for the application even an error has occurred. Also see paragraph 23 of current invention, "Java RMI is a **well-known tool** that can be used to access one JVM from another JVM. Use of Java RMI enables remote invocation and execution of applications and methods. To invoke a remote application, Java RMI creates a thread to the other application or method. Creation of the new thread using Java RMI occurs **without suspending or terminating** the execution application." Since it's already known as a feature of using Java RMI, therefore it's not an

invention.

13. The system recited in claim 9, wherein the object shell console attaches to the application using Java RMI.

For the feature of claim 9 see claim 9 rejection. For the rest of the feature of claim 9, see claim 5 rejection.

17. A system for modifying an executing application in substantially real-time, comprising:

Same as claim 1 rejection.

a computer on which the application is executing;

means for attaching to the executing application so that program data is extracted from the executing application;

a display device for displaying the program data to a maintenance person;

means for accepting a command from the maintenance person; and

means for invoking the command to thereby cause the application to be modified in accordance with the command without suspending or terminating the application.

19. The system recited in claim 17, wherein the attaching means further comprises means for attaching to the application using Java RMI.

For the feature of claim 17 see claim 17 rejection. For the rest of the feature of claim 19, see claim 5 rejection.

11. Claims 6, 14, 18 are rejected under 35 U.S.C. 103(a) as being unpatentable

over U.S. Patent No. 6,662,363 by Hiromu Miyamoto (hereinafter "Miyamoto"), in

view of 'Introduction to java Remote Method Invocation (RMI)', by Chris Matthews

Art Unit: 2122

(hereinafter, "Matthews"), further in view of 'JAVA in A Nutshell' by David

Flanagan (hereinafter "Flanagan").

CLAIM

6. The method recited in claim 1, further comprising the step of modifying an application written in an interpreter programming language.

Miyamoto / Matthews / Flanagan

For the feature of claim 1 see claim 1 rejection. Miyamoto and Matthews teach all aspects of claim 1 but does not mention the 'interpreter programming language' specifically. However, Flanagan teaches that feature in his book. Java is an **interpreter programming language**, see Oreilly, page 14, last paragraph, "To invoke a Java program, you run the Java **interpreter**, java," and page 247, "The **Java Interpreter**" section.

It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Miyamoto and Matthews' disclosure of the method using Java RMI by the feature of Java is an interpreter language further taught by Flanagan, for the purpose of invoking Java RMI (Invoke a Java program, run the Java Interpreter, in page 14).

14. The system recited in claim 9, wherein the application is written in an interpreter programming language.

For the feature of claim 9 see claim 9 rejection. For the rest of the claim 14 features, same as claim 6 rejection.

18. The system recited in claim 17, wherein the application is written in an interpreter programming language.

For the feature of claim 17 see claim 17 rejection. For the rest of the feature of claim 18, see claim 5 rejection.

12. Claims 6 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,662,363 by Hiromu Miyamoto (hereinafter "Miyamoto"), in view of 'Introduction to java Remote Method Invocation (RMI)', by Chris Matthews (hereinafter, "Matthews"), further in view of 'Java Remote Method Invocation' 06/98, by Gopalan Suresh Raj (hereinafter "Raj"), and U.S. 6,681,389 by Norbert Engel et al. (hereinafter "Engel").

CLAIM

7. The method recited in claim 1, further comprising the steps of:
(a) accepting a selection of a method from the program data; and
(b) invoking the method from the command line with at least one new argument.

Miyamoto / Matthews / Engel / Raj

For the feature of claim 1 see claim 1 rejection. Miyamoto and Matthews teach all aspects of claim 1 but does not mention the 'accepting argument' and 'invoking method' specifically. However, Engel teaches the 'accepting a selection' feature and Raj teaches 'invoking method' feature in an analogous art. In Engel, column 4, lines 61-65, "The user can **select** (there must be a list of software **prompted** for the user to do the selection) which application component to upgrade on one or more online machines/servers in the cluster. This provides the flexibility of updating a subset of application software rather than all application software on all machines/servers in a cluster." In Raj, page 1, 2nd paragraph, "In the RMI model, the server defines objects that the client can use remotely. The clients can now **invoke methods of this remote object** (*accepting a selection of a*

method) as if it were a local object running in the same virtual machine as the client. RMI hides the underlying mechanism of transporting method **arguments** and return values across the network." Further, last paragraph, "which in turn **calls the appropriate method** on the server object. In other words, the stub acts as a proxy to the skeleton and the skeleton is a proxy to the actual remote method."

It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Miyamoto and Matthews' disclosure of the method using Java RMI by the feature of entering argument and invoking methods further taught by Engel and Raj for the purpose of passing parameters during method calls between machines (see Raj, page 2, 2nd paragraph).

8. The method recited in claim 1, further comprising the steps of:

- (a) accepting a selection of a method and the program data;
- (b) prompting the maintenance person for at least one new argument value; and
- (c) invoking the method with the at least one argument.

15. The system recited in claim 9, wherein the program data comprises at least one method related that is executed in the application, further

For the feature of claim 1 see claim 1 rejection. For the rest of the features in claim 8, see claim 7 rejection.

For the feature of claim 9 see claim 9 rejection. For the rest of the features in claim 15, see claim 7 rejection.

comprising:

means for accepting a selection of a method from the program data; and
means for invoking the method from the command line with at least one new argument.

16. The system recited in claim 9, wherein the program data comprises at least one method related that is executed in the application, further comprising:

means for accepting a selection of a method from the program data;
a prompt to prompt the maintenance person for at least one new argument value; and

means for invoking the method with the at least one new argument value.

For the feature of claim 9 see claim 9 rejection. For the rest of the features in claim 16, see claim 7 rejection.

20. The system recited in claim 17, wherein the program data includes one or more methods that are executing in the application, further comprising:

means for accepting a selection of a method from the display of the program data; and

means for invoking the method from the command line with at least one new argument.

For the feature of claim 17 see claim 17 rejection. For the rest of the features in claim 20, see claim 7 rejection.

21. The system recited in claim 17, wherein the program data includes one or more methods that are executing in the application, further comprising:

means for accepting a selection of a

For the feature of claim 17 see claim 17 rejection. For the rest of the features in claim 21, see claim 7 rejection.

method from the display of the program data;

a prompt to prompt the maintenance person for at least one argument value; and

means for invoking the method with the at least one argument.

Conclusion

The following summarizes the status of the claims:

35 U.S.C. 103 rejections: 1-21

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chih-Ching Chow whose telephone number is 703-305-7205. The examiner can normally be reached on 7:00am - 3:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on 703-305-4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2122

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Chih-Ching Chow
Examiner
Art Unit 2122

CC



ANTHONY NGUYEN-BA
PRIMARY EXAMINER